

Permutations

Assignment Description

In this assignment you will develop code yourself to be able to work with permutations. In particular, you will then use this to work out compositions of permutations.

The permutation $1 \rightarrow 2, 2 \rightarrow 4, 3 \rightarrow 3, 4 \rightarrow 1$ is written in *cycle notation* as $(1, 2, 4)(3)$ which is usually abbreviated to $(1, 2, 4)$. The composition of two or more permutations is expressed by writing them next to each other. The rightmost permutation is applied first. The composition of $(1, 3, 5, 2)$ and $(1, 6, 3, 4)$ is then expressed as $(1, 3, 5, 2)(1, 6, 3, 4)$. If we work this out from right to left we see that $1 \rightarrow 6 \rightarrow 6, 6 \rightarrow 3 \rightarrow 5, 5 \rightarrow 5 \rightarrow 2, 2 \rightarrow 2 \rightarrow 1$, which gives us the first part gives: $(1, 6, 5, 2)$. Now we want to know what happens to 3 and 4. We find $3 \rightarrow 4 \rightarrow 4$ and $4 \rightarrow 1 \rightarrow 3$, which gives us $(3, 4)$. So we get $(1, 3, 5, 2)(1, 6, 3, 4) = (1, 6, 5, 2)(3, 4) = (3, 4)(1, 6, 5, 2)$.

We do the composition of multiple permutations in the same way.

Write a function `composition` which takes as input a sequence of permutations written in cycle notation and returns as output the composition of the permutations, again in cycle notation.

Simplify the result by ignoring trivial permutations such as (2) . So `func("(2)") = ""`.

Input-Output specification

A single string comprised of multiple permutations written in cycle notation. ie: `"(1,3,5,2)(1,6,3,4)"`

Example

Input	Output
<code>(1,3,5,2)(1,6,3,4)</code>	<code>(1,6,5,2)(3,4)</code>
<code>(2,5)(1,4)(1,2,3)(1,2)</code>	<code>(1,3,4)(2,5)</code>
<code>(1)(2)(3)(1,3,2)</code>	<code>(1,3,2)</code>

Further Reading

Wikipedia: [Permutations](#) and [Permutation group](#)

Hints

Don't put everything in one function, but divide the task into smaller pieces.

Design your code on paper; what features do you need? What is a convenient way to store a permutation? How do you translate the permutation to a string and vice versa?

Don't worry about the order in the output, for example during testing $(1, 3, 4)(2, 5)$ and $(5, 2)(4, 1, 3)$ are equivalent.

Notes

Make sure to turn in your code with the filename `"Permutations.py"`. If you turn it in with any other name it will fail the tests. This is due to instead of this assignment giving you an input and you having to print an output. This assignment will run unit tests on your program itself.

A template for this assignment is provided if you want. If you choose not to use it just do make sure to keep the name of the main function `composition`, otherwise the tests will fail.